

CLAIMS

1. A help architecture in support of an application and a help engine operating on a computer, the help engine for locating help topics relevant to the application, the help architecture comprising:

at least one help library storing a plurality of help topics therein, the help topics being accessible to a user from the help library by way of the computer; and

a central store storing topic metadata corresponding to help topics available from each help library, the central store in response to a search request from the help engine locating topic metadata relevant to the search request and returning information from the located topic metadata.

2. The architecture of claim 1 wherein the search request includes search keywords, each piece of topic metadata includes keywords, and the central store returns information from pieces of topic metadata containing the search keywords.

3. The architecture of claim 2 wherein the returned information corresponding to each piece of topic metadata includes a summary of the corresponding help topic and an address by which the corresponding help topic may be located in the particular help library, whereby the user may review the summary and if desired based thereon access the help topic from the help library.

4. The architecture of claim 1 wherein for each help topic available from a particular help library, the central stores has stored therein

corresponding topic metadata including a summary of the help topic and an address by which the help topic may be located in the particular help library.

5. The architecture of claim 4 wherein for each help topic available from a particular help library, the central store has stored therein corresponding topic metadata 46 further including a set of keywords by which the help topic can be searched for and a set of attributes by which the help topic can be filtered.

6. The architecture of claim 1 wherein each help library includes for each help topic available therefrom a summary of the help topic and an address by which the help topic may be located in the help library.

7. The architecture of claim 1 wherein each help library includes for each help topic available therefrom a set of keywords by which the help topic can be searched for and a set of attributes by which the help topic can be filtered.

8. The architecture of claim 1 wherein each help library has an update interface, the architecture further comprising an update engine for periodically requesting an update of topic metadata from each of the help libraries by way of the update interface thereof, for receiving the update by way of the update interface thereof, and for storing the update in the central store.

9. The architecture of claim 8 wherein the central store includes the update engine.

10. The architecture of claim 8 wherein the update as received by the update engine from the update interface of each help library is a complete set of topic metadata corresponding to all help topics available from the help library.

11. The architecture of claim 8 wherein the update as received by the update engine from the update interface of each help library is a delta set of topic metadata corresponding to all changes to help topics available from the help library since a previous update.

12. The architecture of claim 11 wherein the topic metadata is set forth in a hierarchical format, and the delta set of topic metadata is a hierarchical document set forth in a hierarchical differential language.

13. The architecture of claim 8 wherein the update engine stores the update from each help library serially.

14. The architecture of claim 8 wherein the update engine aggregates the updates from multiple help libraries and then stores the aggregated updates.

15. The architecture of claim 1 wherein the help topics include topics on a member of a group consisting of help, suggested samples, suggested next steps, suggested templates, suggested libraries, and combinations thereof.

16. The architecture of claim 1 wherein the application includes a plurality of components that operate to effectuate application functions, each component of the application being constructed to describe a current context of the component representative of a current state of the component, at least some of the components being active at any one time, the help engine operating to receive the current context from each active component, to dynamically build a current overall context of the application based on each received current context, to locate help topics based on the current overall context, and to filter and prioritize the located help topics.

17. The architecture of claim 1 wherein each help library includes a receive interface, and the help engine includes a complementary post interface for assisting the user in posting a help topic composed thereby to a help library by way of the receive interface thereof, the post interface of the help engine receiving the composed help topic and packaging same and related information in a predetermined format, and sending the packaged help topic 18 to the help library by way of the receive interface thereof, the help library upon receiving the packaged help topic un-packaging and storing as a posted help topic

18. The architecture of claim 17 wherein the help library prior to storing the help topic submitting same for review by a reviewer.

19. The architecture of claim 17 wherein the post interface upon receiving the composed help topic also receives from the user keywords and/or attributes to be ascribed to the composed identified help topic.

20. The architecture of claim 17 wherein the post interface upon receiving the composed help topic also receives from a current context of the application keywords and/or attributes to be ascribed to the composed identified help topic.

21. The architecture of claim 17 wherein the post interface upon receiving the composed help topic also receives from the user additional information relating to the composed help topic including at least one of a title, a summary, and an author.

22. The architecture of claim 17 wherein the post interface upon receiving the composed help topic also receives from the user a selection of at least one help library to receive the composed help topic.

23. A method in combination with an application and a help engine operating on a computer, the method for locating and reviewing help topics relevant to the application, the method comprising:

the help engine sending a search request to a central store and the central store receiving the search request;

the central store based on the search request reviewing topic metadata stored therein and locating pieces of topic metadata relevant to the search request, each located piece of topic metadata corresponding to a help topic available to the user from a help library by way of the computer;

the central store returning information from each located piece of topic metadata, the returned information including for each piece of topic metadata a set of attributes relating to the corresponding help topic and display information;

the help engine filtering each help topic based on the set of attributes thereof and displaying the display information relating to each filtered help topic to the user, the displayed information for each help topic including a summary of the help topic and an address from which the help topic may be obtained from the corresponding help library;

the user reviewing the summary corresponding to each help topic to determine whether the help topic is of further interest; and

if so, the user employing the address of the help topic of interest to obtain such help topic from the corresponding help library.

24. The method of claim 23 wherein the search request includes search keywords, each piece of topic metadata includes keywords, and the central store returns information from pieces of topic metadata containing the search keywords.

25. The method of claim 23 wherein the help topics include topics on a member of a group consisting of help, suggested samples, suggested next steps, suggested templates, suggested libraries, and combinations thereof.